Remailer Networks

Peter Palfrader (ppalfrad@cosy.sbg.ac.at)
Michael Park (mpark@cosy.sbg.ac.at)
Robert Harald Rescher (rrescher@cosy.sbg.ac.at)
Hartmut Wernisch (harti@cosy.sbg.ac.at)

June 12, 2003

What is a remailer?

A remailer is a service that provides anonymous emails, so that the original author can not be retraced. Almost every Internet Service Provider (ISP) can monitor and save any email without knowledge of the users. In many countries ISPs are monitored constantly by government agencies.

A short history

- In the 1980s Karl Kleinpaste established a pseudonymous server for usenet.
- In 1992 the probably most heavy used pseudonymous server was “anon.penet.fi”, established by Johan “Julf” Helsingius.
- 1992 a group of kryptographs formed, called the “Cypherpunks”. Eric Hughes and Hal Finney, two founders of the Cypherpunks developed a anonymous mailing system, the Cypherpunk remailer.
- Lance Cottrell tried to develop a remailer which should improve the Cypherpunk remailer and created Mixmaster.

Reasons why a remailer should be used

Some reasons why you could need a remailer:

- Fear of not being tolerated within a community because of political, religious or social views.
- Ask for help and advice, anonymously, in newsgroups especially in questions of law.
- To stay anonymous within a certain business.
- To be not harassed by advertisements, fans or whatever.
- Nobody wants their email being read by strangers.

Reasons why a remailer should not be used

Some reasons why a remailer could be abused:

- New kind of mass emails that cannot be retraced.
- Safe communication for any kind of criminals and illegal business.
- People can also safely harass other people by email.

The working principle of remailers

It is very important to create a confusing outgoing data stream referrig to data size and latency by sending the encrypted mails to a “Mix” and using an artificial delay.

Currently there are three main remailer systems

- Cypherpunk remailer (Type I)
- Mixmaster (Type II)
- Mixminion (Type III)

Type I – Cypherpunk

- Sending messages to email-addresses or newsgroups.
- Processing the directives of the message.
- Discarding original mail headers.
- Adding new headers.
- Removing specific parts of the message.
- Encrytion/Decrytion of the message.
- Latency.
- Reordering of the messages.

Message Format

- It is possible to compose emails with the standard email-client.
- *Attention:* Disable MIME encoding!

Message format:

::
[Remailer directives]
##
[Hash headers]

[Message text]

Message Example

Example

::
ANON-TO: final@recipient.com
Latent-Time: +1:00
##
Subject: This is an anonymous message example

This ist the text of the message to be remailed to the final recipient specified in the ANON-TO directive.

Using a standard email-client you have to put the address of the remailer into the ’To:’ field and leave the ’Subject:’ empty. Place the message in the body of the mail. It ist very important that the ’::’ is in the first line of the body!

PGP

- It ist highly recommended to use PGP encryption if the remailer supports it!
- **Example of an encrypted email**

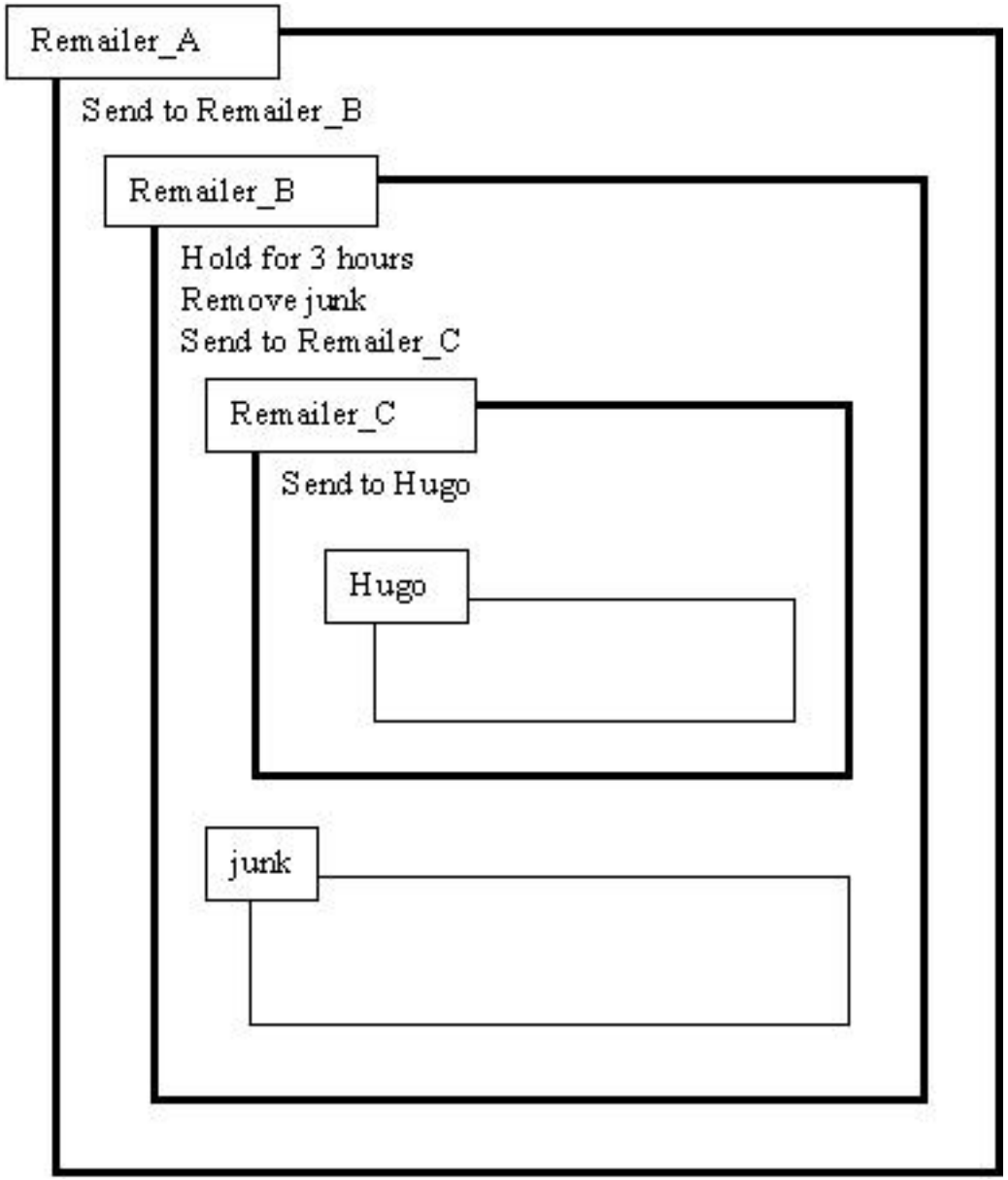
  ::
  Encrypted: PGP

  ——BEGIN PGP MESSAGE——
  Version: 2.6.2

  ahfjkadhflakhfjadf. . .
  ——END PGP MESSAGE——

- Again the blank line after the directive ’Encrypted: PGP’ is mandatory!

Message


Remailer_A
Send to Remailer_B
Remailer_B
Hold for 3 hours
Remove junk
Send to Remailer_C
Remailer_C
Send to Hugo
Hugo
junk

Directives (1)

- TEST-TO:
  Instead of being mailed, a test report is sent to the Test-To address.

- ANON-TO:
  Message is to be remailed anonymously to the specified address.

- ANON-POST-TO:
  Posts message anonymously to the specified newsgroup.

- REMAIL-TO:
  Message is to be remailed anonymously without public key encryption. The message is not RePGPed or Remixed even if transparent repgp or remix is enabled.

- REMIX-TO:
  Instructs to remix the message to the indicated remailer.

Directives (2)

- MAX-DATE / MAX-SIZE / MAX-COUNT:
- CUTMARKS:====
  Indicates that any text after a line containing only ”====”should be discarded.
- INFLATE:
  Adds kilobytes of random garbage to the message.
- LATENT-TIME:
  Schedules message for sending.
- and many more. . .

Problems (Size and Distinguishability)

Size and Distinguishability

You can send your message through a chain of remailers with delaying and reordering at every hop, but your message can still be tracked by size. The fluctuation of the size of a default message is very small and approximately known.

Solution:

- The messages should all be the same size.
- Using *cutmarks* and *inflate*.

Problems (Replay Attack)

Replay attack

Reordered and indistinguishable messages can still be tracked. The replay attack can be used to follow a message to its final destination or to backtrack from the end to the original sender. To trace forward, the attacker captures your message and sends many copies of it to the first remailer. These messages moves from the remailer on to the next remailer. This bump in remailer traffic will show the route of the message.

Solution:

- Refusing copies of a message (using IDs)
- It is hard to prevent such sort of attacks, because you can use reply-blocks several times.

Problems of simple remailer systems

- Dependencies between receiving and sending a message.
  - ⋆ A forward is a brief event.
  - ⋆ To an outgoing message, there must be a preceeding incoming one.
- Traffic.
  - ⋆ The less current traffic, the larger must be the times of latencies to grant same confusion.
- Size of messages.
  - ⋆ A very large message may appear unique in current stream.

Remailer of type 2

Why type-2-remailers?
Let us see the following case:

- Adversary can read input and output of server.
- Adversary can distinguish his messages from the ones of others.

First steps of optimization:

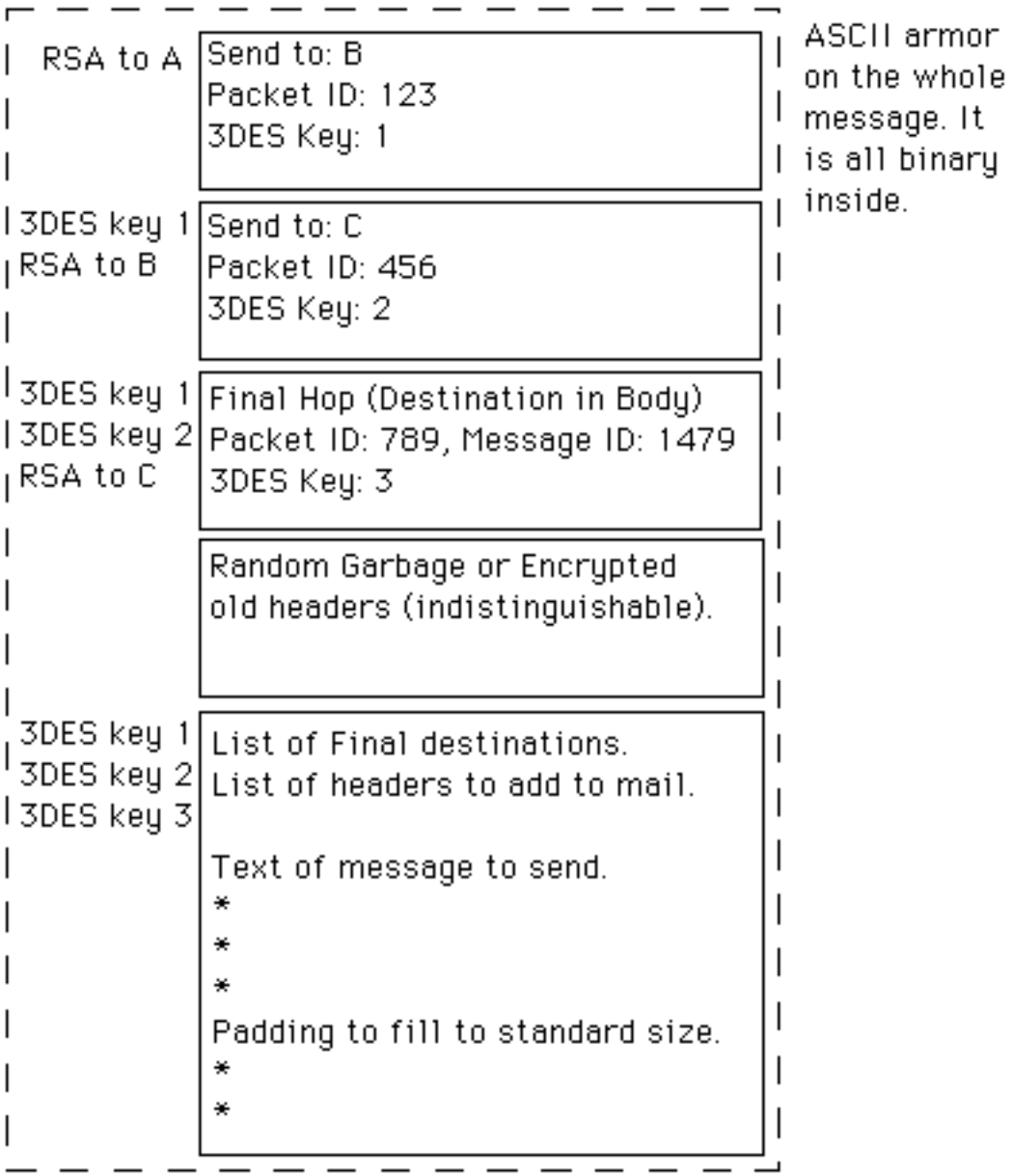
- Delay before forwarding.
- Using a pool.

Name is *Mixmaster*.

Mixmaster messages

- All of them have same size. We may have to:
  - ⋆ pad.
  - ⋆ fragment.
- Encryption:
  - ⋆ Encryption using the algorithms RSA and 3DES.
  - ⋆ One key per remailer.

Mixmaster message format



Not yet perfect

- There always have to be more type-2-servers.
- Dummy-traffic necessary. This must also meet the conditions of usual, real mails.
- Anonymity of receiver is not possible.
- Reply is not possible.

Type III – Mixminion

- Design paper published in the *Proceedings of the 2003 IEEE Symposium on Security and Privacy, May 2003.*
- Reference implementation currently in alpha.
- http://www.mixminion.net/

Subpoena Attacks

Passive subpoena attack

Eve can record messages for later subpoena.

She can also recognise her own messages, which helps other attacks.

Solution: Link encryption with ephemeral keys.

Subpoena Attacks

Passive subpoena attack

Eve can record messages for later subpoena.

She can also recognise her own messages, which helps other attacks.

Solution: Link encryption with ephemeral keys.

Active subpoena attack

Mallory can still record messages from nodes she runs.

Solution: Frequent key rotation.

Partitioning

Partitioning based on client knowledge

Adversaries can distinguish between users based on which directory they use.

Solution: Every client must use the same directory, update at the same time, and use the same algorithm for selecting paths.

Partitioning

Partitioning based on client knowledge

Adversaries can distinguish between users based on which directory they use.

Solution: Every client must use the same directory, update at the same time, and use the same algorithm for selecting paths.

Adversary controls directory

An adversary can publish modified directories, which may favour nodes they control.

Solution: All directory servers need to agree on one directory and each directory server needs to sign the single one list of nodes.

Even more good news

Type I is still needed for Reply Blocks

Since Type II does not support recipient anonymity, Type I is still around.

Solution: Allow for recipient anonymity. Even better: they share the same anonymity set with forward anonymous messages.

Even more good news

Type I is still needed for Reply Blocks

Since Type II does not support recipient anonymity, Type I is still around.

Solution: Allow for recipient anonymity. Even better: they share the same anonymity set with forward anonymous messages.

Fragmentation

In Type II if one fragment of a larger message is lost, the entire message cannot be reassembled at the last hop.

Solution: Mixminion implements a k *out of* n algorithm: Only k of n fragments are required to rebuild the entire message.

Open problems

- Long term intersection attacks
- Trickling attacks
- Dummy messages
- Usability

Any Questions?